Available online at www.sciencedirect.com
ScienceDirect
Procedia CIRP 33 (2015) 115 – 120
www.elsevier.com/locate/procedia

9th CIRP Conference on Intelligent Computation in Manufacturing Engineering - CIRP ICME '14

Flexible and Modular Control and Manufacturing System

Jens Friedrich, Stefan Scheifele, Alexander Verl, Armin Lechler

Institute for Control Engineering of Machine Tools and Manufacturing Units (ISW), University of Stuttgart, Seidenstraße 36, 70174 Stuttgart, Germany

Abstract

The trend in production industry is going away from mass-produced products, towards individual products, which are adapted to the customer requirements. Thus flexible, modular production systems that can be adapted to individual use-cases are necessary. Within this paper, a flexible and modular control for a modular production system with the ability to use manufacturer-independent functions and modules is proposed. A new approach for a standardized description and an open interface for functions and modules is developed. This includes a decentralized control system architecture. The self-configuring control system identifies all functions and modules automatically and a specifically defined memory set (the CIMory data) is used to enable the self-configuration of the control system. Combining a real-time bus system and a SOA-based communication enables the decentralized approach of a future cloud based control system architecture.

Keywords: customized products; flexible and modular production system (FMPS); self-configuring control system

1. Introduction

Modular system-based engineering is the key to a quick, customized machine and plant configuration. This approach is seen as the best method to implement different functional requirements with a minimum amount of resources [1], as well as the possibility to optimize the operational capacity of the whole system, by adding modules to slow production steps [2].

Today’s modular production systems are not designed for online reconfiguration. For a fast reaction on new products, which have low lot sizes because of shorter and shorter product lifecycles, a fast and cost efficient reconfiguration, as well as the adding of functionality, is of huge importance.

2. Modularization of a production system

A flexible modular production system (MPS) is the key for producing companies to react to the trend of individual produced products which are preferred or even needed by their costumers [3, 4]. To achieve this goal of a MPS, able to adapt to a use case, changes in the engineering process compared to the classic machine engineering had to be made. Both, the hardware, as well as the software engineering were developed in the past, but have not achieved the high level of real modularity yet.

In this paper, an approach for a modular control system for a MPS is described. Within the research project CassaMobile a realization mock-up of the MPS is developed [5].

2.1. Modularization of the hardware

The modularization of the hardware components and modules of a MPS are developed so far, that a reconfiguration is possible within a few hours [6]. According to [7] the choice of system boundaries determines the complexity of the module interface, as well as the reconfiguration efficiency. In this context, a module is by definition a sub-system which has a stronger relationship to the inside as to the outside [8]. Module system boundaries should be chosen so that they have as few external interfaces as possible. To achieve this, functions to be performed have to be integrated instead of being added by external periphery. This reduces the number of module interface elements. The module forms a self-contained, autonomous unit.

According to this, modules described in this papers are defined as whole machines which can be used stand-alone but will be combined to an MPS.

doi:10.1016/j.procir.2015.06.022

Another approach for modular and especially micro production system is presented by Hoffmeister et al. [9] for small machine tools for small work pieces. Their production modules are based on a predefined hardware frame onto which either process or kinematic modules can be mounted. This hardware frame in combination to the suitable hardware interfaces of the mentioned modules leads to modularity. The effort for the control software adaption, particularly for the combination of multiple production modules towards an entire production is not considered.

An alternative, but partially similar approach is given by Järvenpää [10] that defines a base module for production modules containing the control cabinet and clean room supply system enabling a workspace with clean room capabilities. To enable a production process, each base module can be equipped with different – but one at a time - production modules (e.g. robot, laser or machining unit). By combining multiple process integrated base modules, an entire production system can be established. This approach ensures a mechanically reconfigurable system, but the automatic adaption of the control system is not taken into account.

2.2. Modularization of the software

Today, all control producers and compatible third-party producers support some elements of the modular design (see product catalogs of e.g., Beckhoff, Bosch Rexroth, Siemens, Wago, Phoenix, Weidmüller, …). They offer control families, which provide a continuous control system. Within these families, the control system manufacturers offer warranty on function. A modular extension within the control family is usually possible (e.g., Siemens SIMATIC S7).

A problem occurs, if a control family is not able to offer a needed functionality. This needed functionality can only be added by applying components of another control family or a functionality of another manufacturer. In many cases the different product families are not compatible and a lot of manual configuration is necessary.

The modularization within the control functions, as described in the project OSACA [11], is not very developed. The exchange of software functionality (OSACA calls this functionality “architectural objects”), is currently not possible. For this reason, control software must be configured depending on the manufacturer of the functionality.

This is one reason why manufacturers of production systems and machines, that have evolved the modularization very far, are caching the configuration and parameter data for the functional module, defined by them, centrally by a manufacturing system configuration system (e.g. EPLAN [12]). By using a kind of bill of materials, the configuration and parameterization of a production system will be generated automatically.

This approach necessarily implies that all needed data for all used functional modules is available in the manufacturer specific plant configuration system. Therefore, appropriate data must be created for new, previously unknown function modules by time consuming work. This includes, among others:

- Configuration and parameterization
- Change options for configuration and parameterization
- PLC and, if needed, CNC program parts
- Information on the development level of intelligence of the function module (either only I/O technology or own intelligence available)
- Sensor processing
- Access protocol for I/O and drives
- Documentation of the function module
- diagnostic possibilities

The machine and system manufacturers require this function module information compatible with their in-house company standard. Thus, a high amount of self-work of the system manufacturers is necessary, which cannot be offered by the function module manufacturer.

An adapter needs to be created (see Fig. 1a, orange, left top to right bottom hatched boxes), which adapts the communication from the standard of the function module manufacturer to the standard of the machine or system manufacturer.

But not only the Control System of a MPS but the whole MPS and its environment can be designed modular. Within the “Industrie 4.0” project, a system is envisaged, in which the function modules bring all necessary information for their use in any control system with them [13]. A service oriented architecture (SOA) is used for communication between manufacturing module and control system [14, 15, 16, 17, 18]. The function modules offer services that are used by the guidance system for the production of the product required. The central plant control systems will continue to develop evolutionary and at the same time the possibilities of decentralized self-organization are increasingly being used [17, 19].

3. Design of a modular production system for plug & use of functions and features

To modularize control functions and to be able to combine them vendor-independent, it is therefore necessary to create a vendor-uniform interface. Function modules from any manufacturer can thus be obtained and integrated into the control system, based on the function performance.

This approach shifts the object of compatibilization from the control system manufacturer to the function module manufacturer (see Fig. 1b).

With this approach it is possible, that the function module manufacturer not only provides the function module, but also the function module information. Thus, the function module is independent from the control system type.

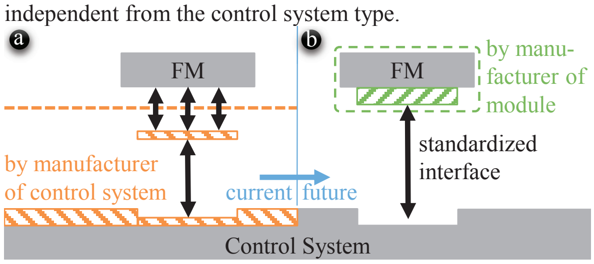

Fig. 1: Current (a) and future (b) integration of function modules (FM) to the Control System by standardization of interfaces

4. Implementation

To implement the new approach, a new modular manufacturing system (MPS) was developed, its control architecture is shown in Fig. 2.

The Workflow Manager (WFM) is the central administrative system within the MPS. Its job is the communication, administration and organization of the individual production modules. It also has the duty to manage and organize the production contracts, as well as the production data management and procurement. The integrated Configuration Manager (CM) of the WFM does the central configuration of all control components.

The production system itself is split into discretized modules which are connected via a communication system which allows a SOA-based communication for production requests, according to the reference model of OASIS [18, 20], and a real-time (RT) based communication for hardware controlling tasks.

In general, a module consists of a mechanic and kinematic structure, the inputs and outputs (I/O) for the sensors and actuators, as well as drive amplifiers (drives).

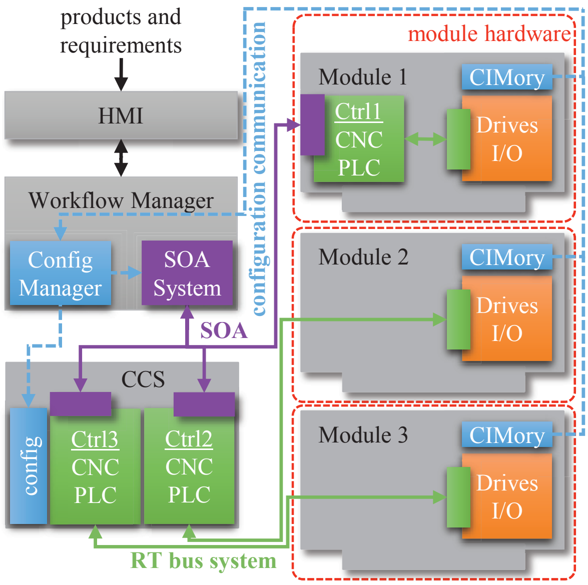

Fig. 2: Overall system architecture

All function modules integrated in the system are self-explanatory and supply their information needed for configuration for the WFM (see Fig. 3).

This information includes a description of the functionality of the module by SOA-services, as well as the needed and supplied data for this functionality. A hierarchical architecture of all possible services of the module is needed to get the whole possible functionality described e.g., “milling”, “3-axis milling” and “5-axis milling” for a 5-axis milling module. This way, the WFM is able to recognize, that a 5-axis milling module can be instructed with a 3-axis task.

With this approach for a standardized and manufacturer-independent self-configuration of a module, it is possible to combine different modules of different manufacturers without the need to create a control system dependent configuration (see Fig. 3). A fast and fail-safe reconfiguration of the production system is therefore possible.

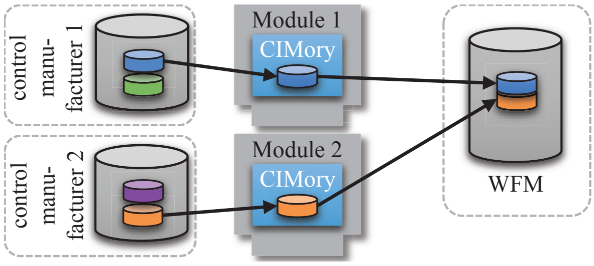

Fig. 3: Configuration over CIMory

For a defined and standardized storage of this module information a unified storage option, called Configuration and Information Memory (CIMory), is required on the function module which is fixedly connected to the function module. The CIMory is designed as a participant on the fieldbus and has a direct relationship to the function module in which it is installed. In addition to the configuration and function memory, it contains a memory area, which gives information about the module type, as well as the unique module ID (see Fig. 4).

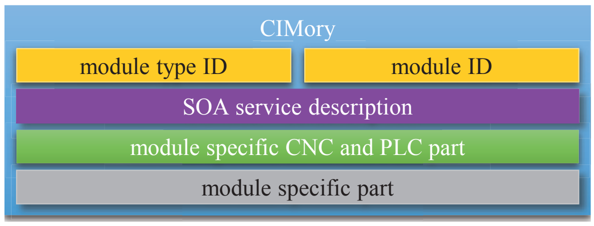

Fig. 4: Internal structure of the CIMory

The access of the CIMory by the Configuration Manager of the WFM is realized by the configuration communication. It will be used to configure and parameterize the CCS-based control systems as well.

5. Sequence of a self-configuration using the CIMory

In the first step of the self-configuration, the WFM initiates a bus scan over the entire production system. This happens in any case during boot up of the production system or by a trigger signal which can be set by the user or a bus system error, which happens by linking or delinking a module.

During the bus scan, all modules register to the WFM with module type ID and module ID and the WFM gets knowledge about all existing modules on the production system.

Thereafter, each of the detected modules will be asked one by one to send its configuration data, which is a standard service of a module. Over the configuration communication, the WFM receives the needed configuration and parameterization of each module. This data packet contains the Module type ID (which type of Function, e.g. milling, does the module provide), the Module ID (a unique ID to address the module), the list and the definition of all provided SOA-services of the module and all other module specific data (see Fig. 4).

Based on this data packet, the CM of the WFM is able to check if a module needs a dedicated control system for operation. If this is the case, the CM collects the needed software-function modules for expanding a standardized base-CNC and base-PLC configuration.

A base-CNC configuration consists of all information needed to run a blank CNC-system. It consists of CNC internal variable definitions, default axis parameters, CNC Chanel definition and parameters and tool data.

Same as the base-CNC configuration, a base-PLC configuration consists of all basic functionality, a PLC-system needs to run. Basic functions (e.g. security functions), which are needed module independent, can be declared in it. If there are global common software I/O for the complete production system, the base-PLC has to define them with ID, name, bit-position in memory and their mapping to hardware I/O.

This runnable base-CNC and base-PLC-Configuration is expanded by module specific parts with information, configuration and functions. In case of the CNC configuration, module specific data is added. Mainly this data includes the axis parameters, which are overwriting the default axis parameters, and CNC channel definitions and parameters, such as the needed axis per channel or the transformation specifying the modules kinematic chain.

For a modular build of the PLC, the PLC configuration consists of sub-programs which are able to read and write hardware I/O and to handle them with if/when/else/and/or commands. Based on this, M-functions (e.g. functions for tool changing), technology functions, process depending functions, module specific security functions and module specific failure prevention is realized.

These PLC parts are, as well as the CNC parts, plain text in standardized manner. Table 1 shows the standard construction of a PLC program:

Table 1. Standard construction of a PLC program.

PROGRAM <*program name*>_<*version*>
(* <comments> *)
VAR
<*definition of variables*>
END_VAR
<*program code*>
END_PROGRAM

This manner enables the configuration process to add PLC parts to a PLC program but requires a vendor-independent framework for PLC program parts. For instance, all used variables in the PLC code are globally valid, which means that they need a standardized rule for naming, as given in Table 2.

Table 2. Exemplary variable naming in a PLC program.

<vendor>_<function_name>_<sub-function_name>_<variable_name>_<vendor_specific_naming>

The CM of the WFM transmits this module specific PLC- and CNC-configuration to the CCS, which implements a new control instance in the CCS and generates a module specific PLC- and CNC-configuration out of basic configurations (see Fig. 5).

After this step, the CCS links the I/O-level and the drives, goes into a safe run mode and checks the validity of the generated PLC- and CNC-configuration. This can happen by a scan over the periphery level for reachable I/O-ports and drives or by a comparison between available periphery (ID of I/O) and configured periphery.

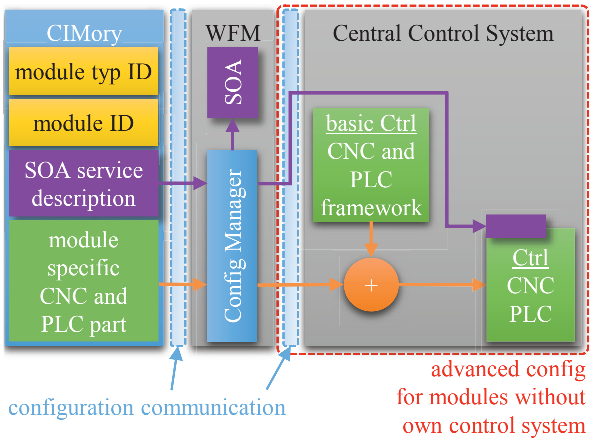

Fig. 5: Configuration communication

After completion of the configuration of all available modules and validation of the deposited and active configuration, the WFM initiates a ramp up of the system. This means the start of the SOA-communication, as well as the start of the control systems for each module.

6. Virtual Modules

In the production chain of today, there are not only the manufacturing steps, but also many software data processing steps to provide the necessary data. In future, the number of production systems with an increasing share of IT-technology will even increase.

To have an automated manufacturing system, the WFM needs IT-data. On the one hand it is necessary for the documentation and traceability of the product, especially for medical products (quality check (QC) and quality management (QM)). On the other hand each process specific data has to be transferred to the process module at the beginning of the process. To simplify the communication, these software tools can be integrated as virtual modules. This means the software components follow the same communication architecture as the hardware process modules. They provide a virtual CIMory with all mandatory data for the configuration of the WFM. The software modules also provide the same common services and the software steps can be directly integrated in the process chain.

From the communication perspective of the WFM there is no difference between a virtual module providing software functionalities and a hardware module manufacturing the part. An example of a virtual module is the CAM process, generating the machine specific G-Code based on a CAD model. Moreover the CCS described before also is a virtual module. As described in Chapter 4, it provides the control software functionalities for the process modules without own control. A scenario using both, virtual and process modules is shown in Fig. 6.

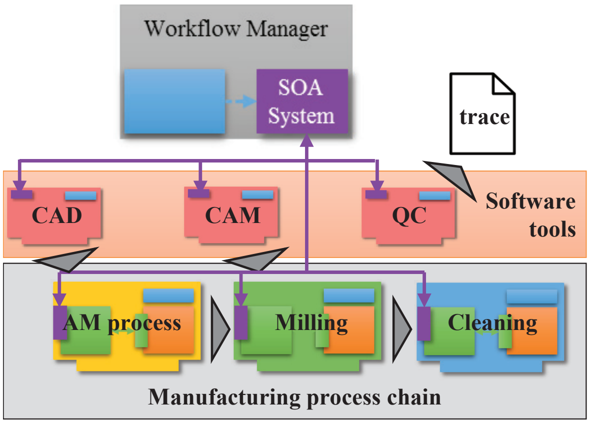

Fig. 6: Process chain including virtual modules and process modules

7. Application-oriented use

The MPS allows a new workflow, which is oriented on the customer's needs. The workflow directly goes from the idea to the product as shown in Fig. 7. The product design is based on the costumer's individual requirements. It can be done by the costumer itself or by an expert providing the service, depending on the business model for the MPS. The design is transferred to the MPS and produced directly on-site at location of need.

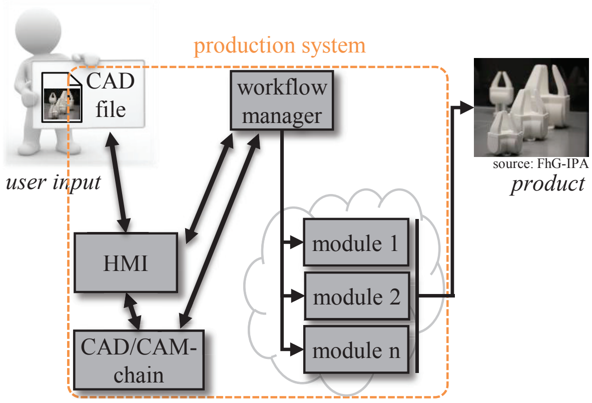

Fig. 7: From CAD to product

There exist several use cases and application domains for which the new container based MPS can be used. In the medical sector the fast reaction on accidents and injuries is essential for an optimal recovery. Today surgeries on fractures in case of accidents are performed without any individual bone drill guides, the use of bone drill guides for difficult fractures is only possible if the surgery is planed several weeks in advanced, because of the long lead time. Using the MPS bone drill guides can be designed based on the x-ray image and produced on-site within few hours using an additive manufacturing (AM) and a milling process. Thus they can be used even in case of accidents and improve the result of the surgery and avoid subsequent surgeries. Another application is for foot orthotics. The bone is measured and the orthotic is produced on-site and any changes for an optimal fitting can be made within short time. But the MPS can also be used in industrial applications. The handling of products is a complex task, depending on the parts geometry. Thus there a new individual grippers necessary, if the product is changed. These grippers can also be produced with the MPS on demand. The different process chains and the configuration of the MPS for these different use cases is show in Fig. 8.

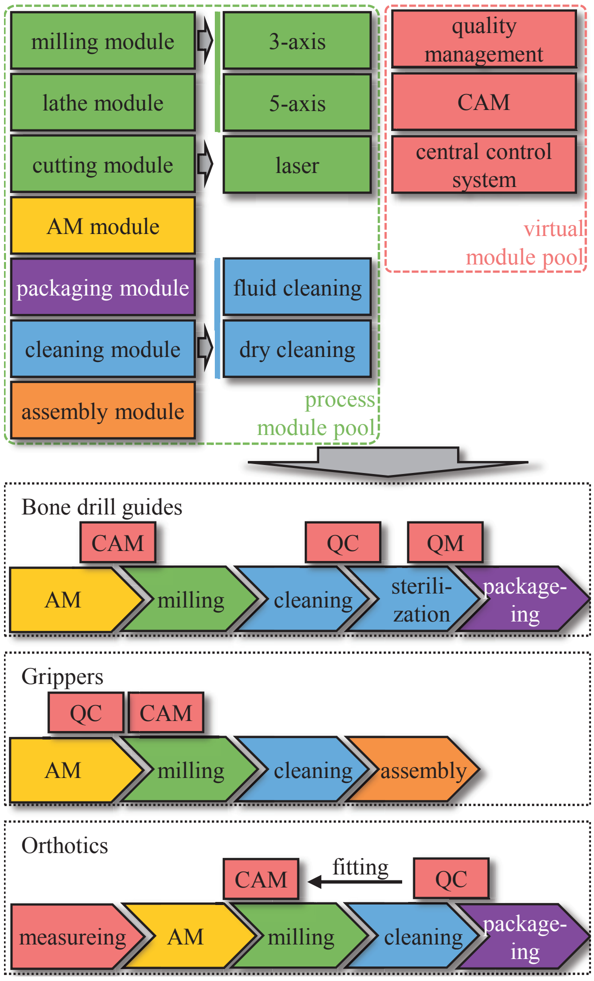

Fig. 8: Use case depending reconfiguration of the production system

With the concept for a self-configuration of a modular production system presented in this paper, it is possible for the manufacturer to quickly and individually respond to use cases and optimally adapt his production systems. Depending on the use case, an individual combination of process modules is possible and enables the modular production system to support multiple different assembly lines.

8. Conclusion

This paper presents an approach for a modular control system which can combine different functions and modules manufacturer independently. The control system reconfigures itself automatically if modules are changed. The control system follows a hierarchical structure with a workflow Manager as master and the controls of each process module as slaves. The non-real-time communication is done by SOA-services.

The SOA-based communication also allows the use of virtual models. This means that software functionalities, like CAM or quality management, necessary for the process chain can also be handled like real process modules. Thus these steps can automatically be integrated in the process chain, by using the same standardized services.

To enable the automated reconfiguration, each module contains an Information and Configuration Memory (CIMory), which contains all necessary information for configuring and parameterizing the module to work with the WFM.

To support the state-of-the-art of industrial practice, the system supports modules with either a highly sophisticated control with internal intelligence or with a simple command executing I/O and drives unit. If there is no control integrated in the module the WFM initiates the configuration of a new virtual, software-based control in the Central Control System for the module. The module specific control information are also provided by the CIMory.

The potential of the modular production system is illustrated on three use-cases in the medical and industrial sector. The use of the MPS enables new application domains of existing product.

Currently the proposed control concept is implemented and the modular production system is built for the explained use-cases. The application of the concept in a cloud-based control system has to be investigated in future research.

Acknowledgements

The research leading to these results was done within the CassaMobile project which has received funding from the European Union Seventh Framework Program under grant agreement n° 609146.

References

[1] Reuter A., Kircher C., Verl A., *Manufacturer-independent mechatronic information model for control systems*, WGP, 2010.

[2] Wang W., Koren Y., *Scalability planning for reconfigurable manufacturing systems*, Journal of Manufacturing Systems, 2012.

[3] Frank T. Piller, Christof M. Stotko (Hrsg.), *Mass Customization und Kundenintegration*, Symposion Publishing GmbH, Düsseldorf, ISBN-10: 3936608059, 2003.

[4] Business Wissen, *Mass Customization - Mit individualisierten Standardprodukten zum Erfolg*, 17.04.2009, accessed 24.04.2014, http://www.business-wissen.de/artikel/mass-customization-mit-individualisierten-standardprodukten-zum-erfolg/.

[5] *CassaMobile project*, funded by the European Union Seventh Framework Program under grant agreement n° 609146, www.cassamobile.eu.

[6] Kircher C, Seyfarth M, Wurst K-H, Pritschow G., *Self-Adapting Control Systems for RMS*. Proceedings of the CIRP-sponsored 3rd International Conference on Reconfigurable Manufacturing, University of Michigan, Ann Arbor, USA, 10.-12. May 2005.

[7] K.-H. Wurst, U. Heisel, C. Kircher, *(Re)konfigurierbare Werkzeugmaschinen – notwendige Grundlage für eine flexible Produktion*, wt Werkstattstechnik online Jahrgang 96 (2006), H. 5, S. 257.

[8] Scholler C., Schindler C., Pick S., Müller S., *Modularer Simulationsbaukasten zur Potenzialabschätzung hydraulischer und hybrider Konzepte*, 4. Fachtagung Hybridantriebe für mobile Arbeitsmaschinen, Karlsruhe, 2013.

[9] Hoffmeister, H.-W. , Gerdes, A. , Verl, A. , Wurst, K.-H. , Heinze, T. , Batke, C., *Investigations of a small Machine Tool with CFRP-frame*, euspen 13th International Conference, Berlin, Germany, pp. 200-203, ISBN 978-0-9566790-2-4, 2013

[10] Järvenpää E., Heikkilä R., Tuokko R., *TUT-microfactory – a small-size, modular and sustainable production system*, GCSM, 2013.

[11] *OSACA (Open System Architecture For Controls Within Automation Systems)*, http://www.osaca.org/.

[12] Pesch D., *Baukasten für disziplinübergreifendes Projektieren*, iee, issue 3, 2004.

[13] Verl A., REVOLUTION INDUSTRIE 4.0, interaktiv, issue 2, FRAUNHOFER IPA, 2012.

[14] Plattform Industrie 4.0, *Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0*, issue 4, 2013.

[15] Bauernhansel T., interview, openautomation, issue 4, 2013

[16] Göhner P., *Agentensysteme in der Automatisierungstechnik*, Springer Vieweg, ISBN 978-3-642-31767-5, 2013.

[17] Cyber-Physical Systems, *Chancen und Nutzen aus Sicht der Automation*, VDI/VDE, issue 4, 2013.

[18] Abel M., Klemm P., *Flexible SOA based Platform for Research on Start-up Procedures for Reconfigurable Production Machines*, FAIM2013.

[19] Post P., Kärcher B., *Basistechnologien für innovative Konzepte*, vdma-Nachrichten März 2013.

[20] OASIS, *Reference Architecture Foundation for Service Oriented Architecture*, Version 1.0, 2012.